2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T) | 978-1-7281-9177-5/20/$31.00 ©2020 IEEE | DOI: 10.1109/PICST51311.2020.9467963

Concept of Intelligent Detection of DDoS Attacks in SDN Networks Using Machine Learning

Mykhailo Klymash, Olga Shpur, Nazar Peleh
Department of Telecommunication
Lviv Polytechnic National University
Lviv, Ukraine
mykhailo.m.klymash@lpnu.ua, o.shpur@gmail.com,
van_plus_k@ukr.net

Oksana Maksysko
Faculty of Food Technologies and Biotechnology
Stepan Gzytsky Natoinal University of Veterinary
Medicine and Biotechnologies
Lviv, Ukraine
oksana.maksisko@i.ua

***Abstract* — In this paperwe propose the concept of intelligent detection of DDoS attacks in SDN networks by log analyzing. Due to SDN management and implementation of the self-learning element, we propose to teach the SDN controller to detect attacks using information about the state of the flow, the duration of the session and its source, using information from logs and flow tables. To do this, it is necessary to divide the total traffic flow into anomalous and normal. Realizing which client requests are the result of DDoS-attack, one can create the appropriate rules for their blocking. We propose to do this by determining the metrics of traffic behavior using the Kulbak-Leibler approach to detect flow anomalies over the session time. In our case, we will compare the average session time with time to access the server from specific IP addresses. The obtained values will be recorded in the Machine Learning database. If the result of the comparison did not bring results, the duration of access to the service during the last seven days is compared. Similarly, the value of KL is determined and written to the ML database. KL accumulation values in a ML will identify anomalies in the flow admission requests by analyzing the length of service and access to prescribe rules of Controller. As a result of using machine learning, the SDN controller will block IP domains from which DDoS attacks are just starting**

Keywords—DDoS atack; SDN; log analysis; machine learning; SDDC

I. Introduction

Management of any network infrastructure is inevitably related to the problem of increasing the security of data from unauthorized access. Data loss or unauthorized use can cost companies big money. This requires some monitoring system that will analyze potential problems that may arise during the work, and run protection system in the service or to notify administrators about possible threats. These systems have a very short time to track changes that occur with the system, e.g., various injection, attacks, technical failures in which the service ceases to function normally.

The effective method to track and prevent this type of problem is control over the time each session is set up and analysis of the logged service information. Such analysis should be carried out at the network core for all login requests as soon as an incoming packet passes the firewall network and is routed to Black Hole.

II. Motivation and Related Work

The use of log data to analyze and monitor the efficient and reliable operation of IT systems and services is the subject of many research papers. Scientists Hu Q., Tang, B., and Lin, D. automated the analysis of data with the division of log content into classes. The results of their research are presented in [1]. Each event in the history of the journal was assigned its own weighting factor, which was determined by the correlation between the number of unique user requests during the hour of its activity in the system, observed for n days. Thus, events were classified as anomalies (when the value of the coefficient was greater than the average over time t) and normal.

Researchers Max Landauer et al [2] suggest detecting anomalous behavior based on grouping rows of logs by similarity to establish static cluster maps. The rows of logs are divided into existing cluster maps created in the previous and subsequent time windows. This establishes a link between clusters of two adjacent cluster maps that previously did not have common elements. This allows to calculate the overlap metric, which determines the probability of transition from cluster to cluster. However, at each subsequent stage of the system, additional clustering is performed to create new static maps.

The solution to such problems could be the introduction of machine learning, especially in the framework of the software-defined data center (SDDC). In paper [3] authors present a secure framework for IoT based on SDN with a brief review of the security in SDN architecture. They also present a ML-based IDS. It uses deep learning with a Restricted Boltzmann Machine (RBM). For simulation, the authors focused on the detection model with Tensorflow and used KDD99 dataset. The proposed algorithm showed 94% of accuracy.

Authors in [4] present a proposal for both IDS and an action triggered by it: Moving Target Defense. They created a simulated network to obtain data for the training (about 40 000 packets). Regarding the architecture, they presented a neuroevolutionary model as a light weight detector that allows real-time operation. To achieve it they developed two distinctive detectors, one per each type of attack: DDoS and worm. To combine the detectors authors use Neuroevolution of Augmenting Topologies (NEAT), an approach to neuro-evolution with crossover context.

There are also proposals for specific network scenarios. That is the case of [5] that presents the implementation of ML-based IDS in optical SDN, and **Ошибка! Источник ссылки не найден.** that proposes a scenario of Intelligent Transport Networks.

Lin and Wang **Ошибка! Источник ссылки не найден.** proposed a DDoS attack detection and defense

978-1-7281-9177-5/20/$31.00 ©2020 IEEE

mechanism based on SDN, but the method used three Openflow management tools with sFlow standard to perform anomaly detection, so the deployment and operation are complex. Yang et al. **Ошибка! Источник ссылки не найден.** proposed a method in which the flow information and the IP entropy characteristics are combined, which has a higher and more accurate detection effect. Although information entropy is flexible and convenient, it still needs to be combined with other technologies in determining the threshold and multielement weight distribution. Saied et al. **Ошибка! Источник ссылки не найден.** advanced that based on analysis of the characteristics of TCP, UDP and ICMP protocols through training ANN algorithm to detect DDoS attacks. The method determines packet protocol which is complex and inefficient.

In [9], the SOM algorithm is used to detect DDoS attacks by extracting the flow statistics related to DDoS attacks. This method has the characteristics of low consumption and high detection rate. The key point lies in the extraction of time interval. The disadvantage of this method is that the detection has a certain hysteresis and the attack behavior is not timely and accurately found. In **Ошибка! Источник ссылки не найден.**, the authors proposed a framework for detection and mitigation of DDoS attacks in a large-scale network, but it is not suitable for small-scale deployment. In [11], authors propose a DDoS attack detection mechanism based on the database of legitimate source and destination IP addresses . The mechanism uses nonparametric cumulative algorithm CUSUM. It analyzes the abnormal characteristics of source and destination IP address when the DDoS attack occurs and effectively checks the DDoS attack, but the method needs to adjust and determine the threshold.

All the studies analyzed above are based either on statistical methods of training security systems or require additional analysis of network traffic. In [13], we also investigated how to monitor web application availability and detect DDoS attacks based on log analysis. In this paper, we would like to improve the previously proposed concept by introducing an element of machine learning and SDN management.

III. Concept of Intelligent Detection of DDoS Attacks in SDN Networks Using Machine Learning

Preferably, all systems that work with web applications have software control. In the SDN architecture the data streams pass through OpenFlow switches forming flow tables. SDN controller transmits, manages and collects statistical information by searching for records in group flow tables from one or several interfaces or applications. Such tables contain information about the sources of the network origin of the streams and the type of interfaces used for transmission. When it comes to applications, SDN controller redirects request flows to the desired application. In this case, the controller will be responsible for the security of these applications.

Most often the web applications are subject to DDoS attacks as the failure of some of the applications can lead to loss of service and its unavailability for consumers.. There is currently no universal tool to counteract DDoS attacks. To counter distributed denial-of-service attacks, there are two main tasks to be followed:

1. Detect DDoS attack as soon as possible.

2. Divide traffic flow into anomalous and normal. Realizing which client requests are result of DDOS-attack, one can create the appropriate rules for their blocking.

To solve the first task, we propose using machine learning: to teach the SDN controller to detect attacks using information about the state of the flow, the duration of the session and its origin. This information can be retrievedfrom the tables of flows.

The second problem we propose to solve by determining metrics of traffic behavior using the Kulbak-Labler approach to detect flow anomalies over the session time. The basis of the proposed concept is shown in Fig. 1

As soon as requests for access to web applications are received by Openflow switches, information about them is sent to the controller and logged in. Since constant reading from the file will be quite costly in terms of resources, it was decided to transfer the logs to the Log Analysis Subsystem. This will speed up the search and filtering needed records. The principle of writing/reading a log to the Log Analysis Subsystem is the same as in [13]. We will format the log as following:

$$'ip\_address--[``yyyy`/`MM`/`dd`:`HH`:`mm`:`ss``]` \quad (1)$$

To recognize a Dos attack in [13], an algorithm was developed that allows to distinguish the user among others as one who attacks the service. To do this, a request is sent to the Log Analysis Subsystem, which finds all calls to the web service in the period from 24 hours before the request and until the moment of the same request. In the dedicated module we separate the maximal number of requests for each individual IP address. Among all requests received within 24 hours, there is an IP address from which the largest number of requests was sent. To verify that the maximum number of requests does not differ from the number of requests from other users, we find the average number of requests M_{23} among all other users, except for the number of maximum requests. We will compare whether the maximum number of requests will be greater than the average, taking into account the correction coefficient. This correction coefficient was chosen as number 10, as it allows to cut off bursts of traffic in peak load hours from real DDoS attacks, where a large number of requests are sent from a single IP address. Since this will stand out among the total web traffic, it will be possible to determine the IP address from which the Dos attack is likely to be carried out. In addition, an important parameter for us is the time of this session.

Let's define the time to access the web server $\{T_{access}\}$ from $\{P\}$ multiple IP addresses. Based on this information we will learn system. To do this, we determine the relative entropy of Kulbak-Labler.

Kullback-Leibler (KL) divergence is a measure of how one probability distribution differs from the reference one. This includes relative (Shannon) entropy in information systems, randomness in continuous time series, and the increase in information when comparing statistical models of inference. In contrast to the variety of information, it is an asymmetric inter-distribution measure, and therefore does not meet the requirements of the statistical scatter metric. In the simplest case of zero divergence Kullback–Leibler show that deals with the points two distributions are identical.

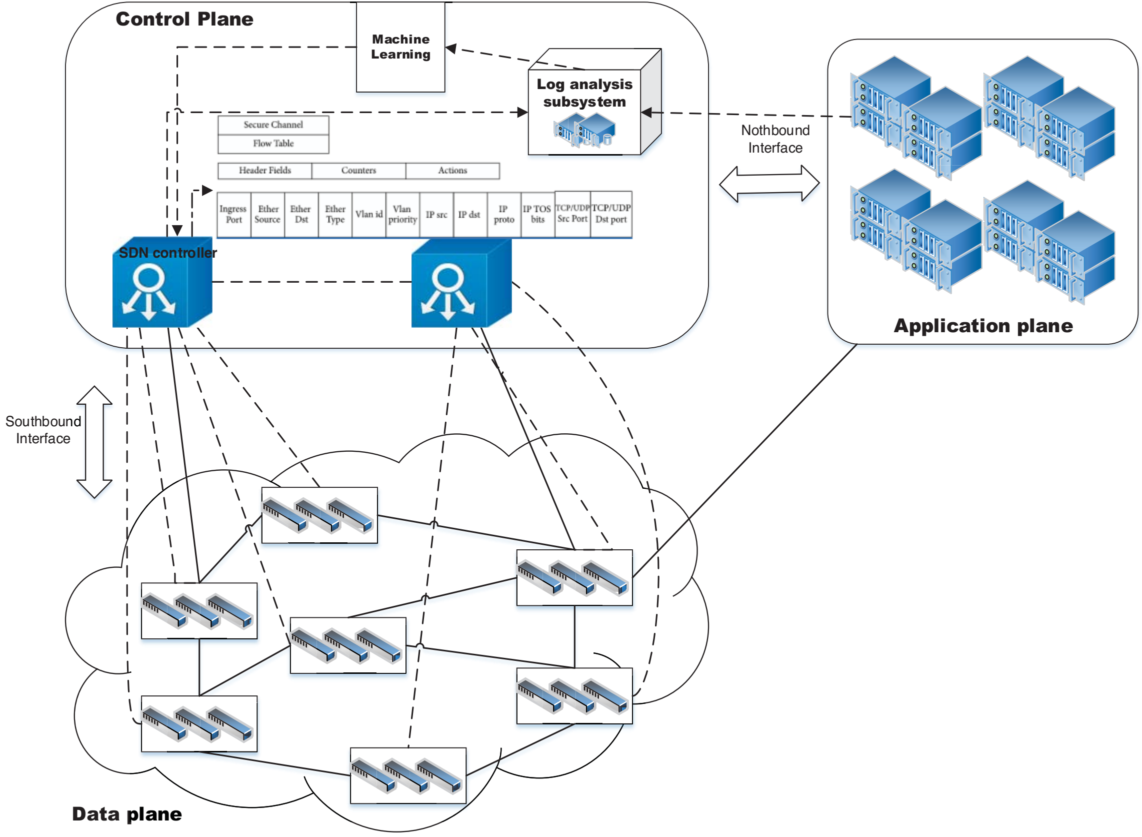

Fig. 1. The proposed concept of SDN architecture with Ddos attacks detection

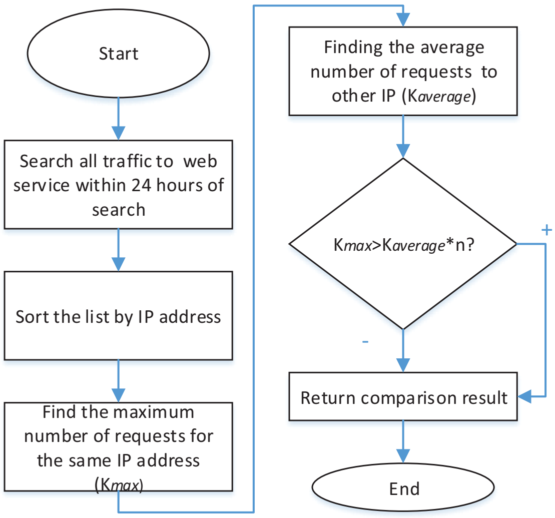
Start
Search all traffic to web service within 24 hours of search
Sort the list by IP address
Find the maximum number of requests for the same IP address (Kmax)
Finding the average number of requests to other IP (Kaverage)
Kmax>Kaverage*n?
+
-
Return comparison result
End

Fig. 2. Traffic analysis algorithm of the DDoS attack [13]

In our case, we will compare the average session time with the time of access to the server from specific IP addresses, which were sorted as a result of the algorithm presented in [13]. To determine KL, we need to calculate the information entropy of both distributions and find their difference. For our case it will be determined as:

$$KL(T_{aver} \,||\, T_{access\_last\_hour}) = \sum T_{aver}(P) \log \frac{T_{aver}(P)}{T_{access\_last\_hour}(P)} \quad (2)$$

The obtained values will be recorded in the Machine Learning database. If the result of the comparison did not bring results, the time of access to the service during the last seven days is compared. Similarly, the value of KL is determined and written to the ML database. The accumulation of KL values in the ML database will allow to detect anomalies in the request flows, based on the analysis of access time to the service and prescribe the rules of the controller. As a result of using machine learning, the SDN controller will block IP domains from which DDoS attacks are just starting. The general scheme of this algorithm is shown in figure 3.

IV. Conclusion

In this article we continue to explore the availability of web services in software-defined networking and detecting/predicting DDoS attacks based on log analysis. Due to SDN management and implementation of the self-learning element, we propose to teach the SDN controller to detect attacks using information about the state of the flow, the duration of the session and its source, using information from logs and flow tables. To do this, it is necessary to divide the total traffic flow into anomalous and normal. Realizing which client requests are the result of DDOS-attack, we can create the appropriate rules for their blocking. We propose to do this by determining the metrics of traffic behavior using the Kulbak-Leibler approach to detect flow anomalies over the session time. In our case, we will compare the average session time with time to access the server from specific IP addresses. The obtained values will be recorded in the Machine Learning database. If the result of the comparison did not bring results, the duration of access to the service during the last seven days is compared. Similarly, the value of KL is determined and written to the ML database. KL accumulation values in a ML will identify anomalies in the flow admission requests by analyzing the length of service and access to prescribed rules of controller.

As a result of using machine learning, the SDN controller will block IP domains from which DDoS attacks are just starting.

This paper offers only a concept. In further research we plan to prove the effectiveness of the proposed concept.

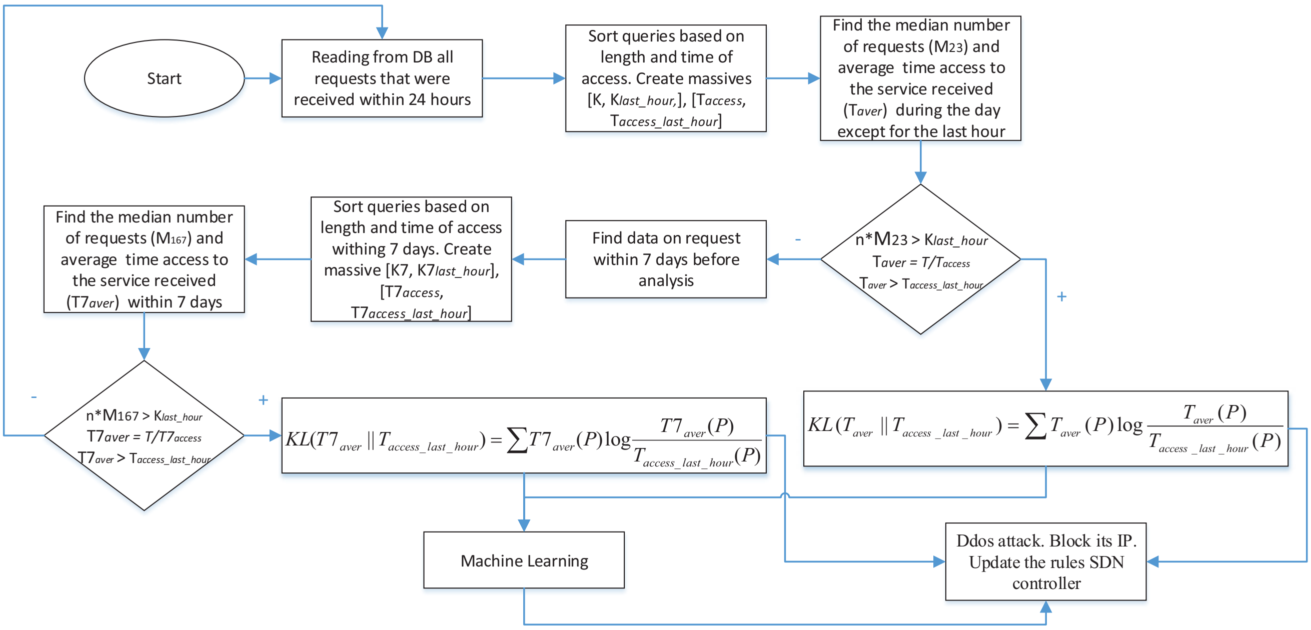
Start
Reading from DB all requests that were received within 24 hours
Sort queries based on length and time of access. Create massives [K, Klast_hour,], [Taccess, Taccess_last_hour]
Find the median number of requests (M23) and average time access to the service received (Taver) during the day except for the last hour
n*M23 > Klast_hour
Taver = T/Taccess
Taver > Taccess_last_hour
-
+
Find data on request within 7 days before analysis
Sort queries based on length and time of access withing 7 days. Create massive [K7, K7last_hour], [T7access, T7access_last_hour]
Find the median number of requests (M167) and average time access to the service received (T7aver) within 7 days
n*M167 > Klast_hour
T7aver = T/T7access
T7aver > Taccess_last_hour
-
+
$KL(T7_{aver} \| T_{access\_last\_hour}) = \sum T7_{aver}(P) \log \frac{T7_{aver}(P)}{T_{access\_last\_hour}(P)}$
$KL(T_{aver} \| T_{access\_last\_hour}) = \sum T_{aver}(P) \log \frac{T_{aver}(P)}{T_{access\_last\_hour}(P)}$
Machine Learning
Ddos attack. Block its IP. Update the rules SDN controller

Fig. 3. The algorithm that recognizes the Ddos attack

REFERENCES

[1] Q. Hu, B. Tang and D. Lin, "Anomalous user activity detection in enterprise multi-source logs", in *Proc. IEEE Int. Conference on Data Mining Workshops (ICDMW)*, New Orleans, LA, USA, 2017, pp. 797-804.

[2] M. Landauer, M. Wurzenberger, F. Skopik, G. Settanni and P. Filzmoser, "Dynamic log file analysis: An unsupervised cluster evolution approach for anomaly detection", *Computers & Security*, vol. 79, pp. 94-116, 2018. Available: 10.1016/j.cose.2018.08.009.

[3] A. Dawoud, S. Shahristani and C. Raun, "Deep learning and software-defined networks: Towards secure IoT architecture", *Internet of Things*, vol. 3-4, pp. 82-89, 2018. Available: 10.1016/j.iot.2018.09.003.

[4] R. Smith, A. Zincir-Heywood, M. Heywood and J. Jacobs, "Initiating a Moving Target Network Defense with a Real-time Neuro-evolutionary Detector", in *In: Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion*, New York, New York, USA, 2016, pp. 1095–1102.

[5] H. Zhang, Y. Wang, H. Chen, Y. Zhao and J. Zhang, "Exploring machine-learning-based control plane intrusion detection techniques in software defined optical networks", *Optical Fiber Technology*, vol. 39, pp. 37-42, 2017. Available: 10.1016/j.yofte.2017.09.023.

[6] A. Raj, T. Truong-Huu, P. Mohan and M. Gurusamy, "Crossfire Attack Detection using Deep Learning in Software Defined ITS Networks", in *Proceedings of 89th Vehicular Technology Conference (VTC2019-Spring)*, Kuala Lumpur, Malaysia, 2019. Available: 10.1109/VTCSpring.2019.8746594

[7] H. Lin and P. Wang, "Implementation of an SDN-based security defense mechanism against DDoS attacks", in *Proceedings of the 2016 Joint International Conference on Economics and Management Engineering (ICEME 2016) and International Conference on Economics and Business Management (EBM 2016)*, Pennsylvania, Penn, USA, 2016. Available: 10.12783/dtem/iceme-ebm2016/4183

[8] J. G. Yang, X. T. Wang, and L. Q. Liu, "Based on traffic and IP entropy characteristics of DDoS attack detection method", *Application Research of Computers,* vol. 33, no. 4, pp. 1145–1149, 2016.

[9] A. Saied, R. Overill and T. Radzik, "Detection of known and unknown DDoS attacks using Artificial Neural Networks", *Neurocomputing*, vol. 172, pp. 385-393, 2016. Available: 10.1016/j.neucom.2015.04.101.

[10] R. Braga, E. Mota and A. Passito, "Lightweight DDoS flooding attack detection using NOX/OpenFlow", in *Proceedings of the 35th Annual IEEE Conference on Local Computer Networks (LCN '10)*, Denver, Colo, USA, 2010, pp. 408–415.

[11] N. Bawany, J. Shamsi and K. Salah, "DDoS Attack Detection and Mitigation Using SDN: Methods, Practices, and Solutions", *Arabian Journal for Science and Engineering*, vol. 42, no. 2, pp. 425-441, 2017. Available: 10.1007/s13369-017-2414-5.

[12] X. Wang, M. Chen, C. Xing, and T. Zhang " Defending DDoS attacks in software-defined networking based on legitimate source and destination IP address database", *IEICE Transaction on Information and Systems*, vol. E99D, no. 4, pp. 850–859, 2016.

[13] M. Klymash, N. Peleh, O. Shpur and S. Hladun, "Monitoring of Web Service Availability in Distributed Infocommunication Systems", in *Proceedings of the 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET-2020)*, Lviv-Slavske, Ukraine, 2020, pp. 723-728.